Niveau(x) d'enseignement: 0 Numéro EDUQ : 9270794
Centre d'intérêt: 640

La Palme, Jean-Baptiste.

Le langage naturel comme paradigme dans l'expression algorithmique des problèmes en robotique éducative / par Jean-Baptiste La Palme. -- Montréal : Université du Québec à Montréal, Centre interdisciplinaire de recherche sur l'apprentissage et le développement en éducation, 1991.
10 f. : ill. ; 29 cm. -- (Séminaire sur la représentation ; 59)

"La publication de ce séminaire est rendue possible grâce à une subvention du fonds FCAR". Notes bibliogr.: f. 10.

Vedette(s)-secondaire(s): Université du Québec à Montréal. Centre interdisciplinaire de recherche sur l'apprentissage et le développement en éducation

Descripteur(s): Robotique * Langage * Utilisation de l'ordinateur * Algorithme

Plusieurs approches ont été retenues à ce jour pour traiter le langage naturel à l'aide de l'ordinateur. L'approche qui est ici favorisée permet à l'enfant de créer son propre langage, avec le moins de contrainte informatique possible, tant sémantique que syntaxique. Ce langage lui permet par la suite de résoudre certains problèmes de robotique éducative. L'expression algorithmique de la solution des problèmes à l'aide de ce langage devrait donner des indices sur la représentation mentale des objets sous considération.

Type de document:

9270794

CENTRE INTERDISCIPLINAIRE DE RECHERCHE SUR
L'APPRENTISSAGE ET LE DÉVELOPPEMENT EN ÉDUCATION

cirade

Université du Québec à Montréal

9270794

Le langage naturel comme paradigme dans l'expression algorithmique des problèmes en robotique éducative

par Jean-Baptiste La Palme
Département de mathématiques et d'informatique
et Maurice Bélanger, CIRADE
Université du Québec à Montréal

Séminaire sur la Représentation

26 janvier 1990 - n° 59

CIRADE, UQAM, juin 1991

Note: La publication de ce séminaire est rendue possible grâce à une subvention du fonds FCAR.

RÉSUMÉ

Plusieurs approches ont été retenues à ce jour pour traiter le langage naturel à l'aide de l'ordinateur. L'approche que nous avons retenue permet à l'enfant de créer son propre langage, avec le moins de contrainte informatique possible, tant sémantique que syntaxique. Ce langage lui permet par la suite de résoudre certains problèmes de robotique éducative. L'expression algorithmique de la solution des problèmes à l'aide de ce langage devrait nous donner des indices sur la représentation mentale des objets sous considération.

Notre approche

Pour illustrer l'approche que nous avons choisie, étudions la situation de robotique éducative suivante (Figure 1): un véhicule mû par un moteur est équipé de deux interrupteurs visant à informer l'ordinateur de la proximité d'un obstacle (Exemple tiré de La Palme 1990). Nous demandons à l'élève de contrôler ce véhicule.

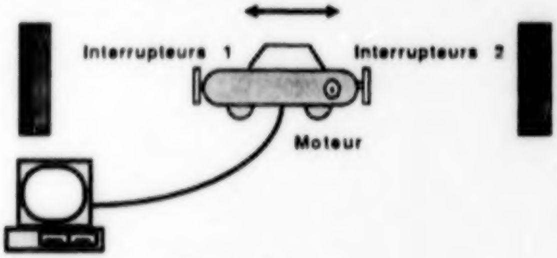


Figure 1

Le problème pourrait s'exprimer ainsi:

Contrôler l'automobile de façon à ce qu'elle change de direction lorsqu'elle frappe un obstacle et qu'elle s'arrête si les deux pare-chocs sont renfoncés.

La façon classique de chercher la solution du problème, en informatique, est de traduire cet énoncé dans un autre énoncé plus proche de son expression algorithmique. Le nouvel énoncé pourrait être celui-ci:

Au départ, l'automobile avance.
Si elle frappe à l'avant, elle recule.
Si elle frappe à l'arrière, elle avance.
Si elle frappe à l'avant et frappe à l'arrière, elle arrête.

Cet énoncé, qui est formé de phrases en langue naturelle, mais qui est très près de la solution algorithmique du problème est souvent appelé **pseudo-programme.**

Jusqu'ici, la réflexion et les énoncés pourraient être exactement les mêmes, quelque soit le langage informatique dans lequel nous avons l'intention de traduire le problème: que ce soit BASIC, Pascal, LOGO, FORTH ou C.

Mais lorsque nous tentons de traduire l'algorithme précédent dans un des langages précités, les problèmes que nous rencontrons sont des problèmes qui ont trait au langage informatique choisi, et non pas au problème sous considération. Cette dernière étape de traduction dans un langage informatique est souhaitable si le but de l'exercice est d'apprendre la programmation dans ce langage. Mais autrement, cette dernière étape est inutile.

Nous nous sommes donc posés les questions suivantes:

Un programme est-il nécessaire?
Pourquoi le pseudo-programme ne serait-il pas le programme?

Ce langage naturel algorithmique aurait comme premier avantage de forcer l'enfant à trouver une expression algorithmique de son problème, et comme deuxième avantage de permettre à l'enfant de vérifier immédiatement sa solution à l'ordinateur.

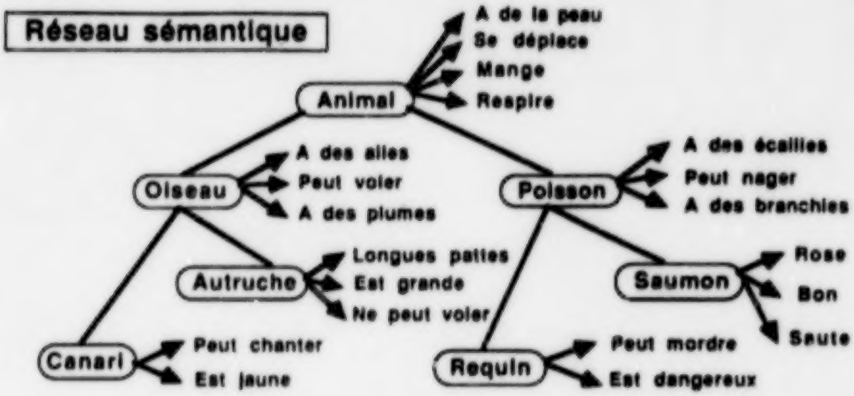
C'est ce langage naturel algorithmique que nous avons réalisé. Cependant, nous sommes allés plus loin. En effet, puisque le langage est indissociable du corpus notionnel qu'il exprime, ne serait-il pas souhaitable que l'enfant puisse définir son propre langage.

Les analyseurs

La «compréhension» du texte par un ordinateur se divise en deux grands champs d'étude. D'une part, les analyseurs et d'autre part les interpréteurs et les compilateurs.

Les analyseurs sont appelés en anglais les «parser». Ils visent à faire une analyse sémantique ou syntaxique du texte, ou les deux.

L'analyse sémantique est généralement basée sur une certaine représentation sémantique du domaine et on peut citer à ce chapitre les réseaux sémantiques (Figure 2), l'analyse conceptuelle ou les «frames» de Minsky.



Traduit de «Applied Natural Language Processing» Steven C. Shwartz p.46

Figure 2

Par ailleurs, l'analyse syntaxique est généralement basée sur une représentation syntaxique ou sur une «grammaire». On peut citer les réseaux de transition (Chomsky) (Figure 3), la grammaire transformationnelle, la grammaire fonctionnelle et la «Case grammar».

Les deux types de représentation précédents se combinent dans des «langages de représentation des connaissances» et les «systèmes experts» qui sont utilisés pour créer à l'intérieur de l'ordinateur une représentation la plus fidèle possible du texte, dans le but d'une action éventuelle ou d'une requête d'information (lire à ce sujet, Winograd 1983 et Bobrow 1977).

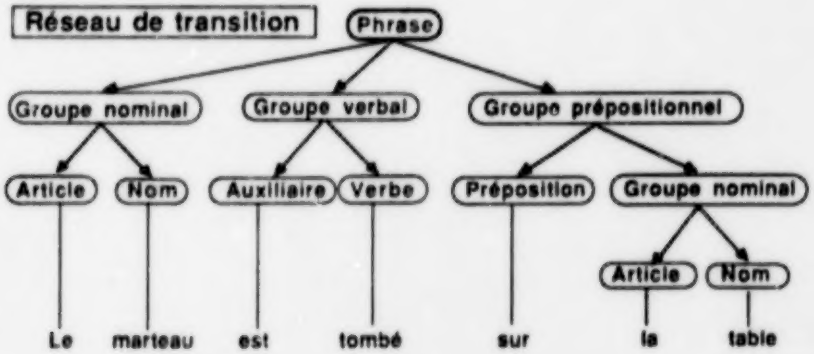


Figure 3

Dans ce champ d'étude, on entendra souvent dire que l'ordinateur «comprend» le texte. En fait, ce qui est impliqué par cette expression, c'est que l'ordinateur possède une représentation interne du texte qui est différente du programme qui fera une action ou qui répondra à une requête d'information: la représentation interne du texte n'est pas une représentation procédurale.

Les traducteurs

L'autre grand champ d'étude a comme principaux représentants les interpréteurs et les compilateurs.

Les interpréteurs et les compilateurs sont des types particuliers de traducteur qui, à partir du texte, produisent un programme dont le fonctionnement et la logique sont davantage calqués sur le fonctionnement et la logique internes de l'ordinateur. Ces traducteurs déterminent ce qu'on appelle des «langages».

Plusieurs types de langages existent. Certains langages peuvent, en théorie, permettre la solution de n'importe quel problème informatique. On les appelle des langages de programmation. D'autres langages ont une sémantique et une syntaxe plus fermées, mais permettent la solution plus rapide d'un certain type de problème. On les appelle des langages d'application.

D'autres enfin, pour en rester à ceux auxquels nous voulons faire allusion, ont une forme naturelle limitée et permettent l'interrogation des bases de données. On les appelle des langages d'interrogation. On pourrait caractériser certains de ces langages comme des langages algorithmiques naturels, en ce sens qu'à la lecture, un programme ressemble au langage naturel, tandis qu'à l'écriture ce langage requiert une activité mentale de création d'un algorithme.

Pour permettre d'appliquer les langages d'interrogation à des champs nouveaux, certains systèmes donnent accès à une grammaire sémantique. Cette grammaire sémantique permet d'intégrer dans le langage d'interrogation des mots et des «pattern» plus près du domaine d'application cible. Le produit se rapproche alors davantage ce que l'on a appelé un langage d'application.

Positionnement de notre recherche

Notre recherche se situe dans le champ d'étude des traducteurs, et comporte deux volets.

Premièrement, nous sommes intéressés à la façon dont les enfants utilisent un langage d'application en vue de la solution d'un problème de robotique éducative.

Deuxièmement, nous sommes intéressés à la façon dont un enfant peut définir son propre langage d'application, à l'aide d'une grammaire sémantique, à partir d'un langage algorithmique naturel.

Jusqu'à ce jour, les recherches concernant les enfants se sont situées à deux niveaux.

Au niveau des analyseurs, beaucoup de recherches ont porté sur la «compréhension» du texte d'un enfant dans le but de le corriger, ou dans le but de lui permettre de communiquer avec une banque de connaissances, ou simplement de communiquer avec un didacticiel. Certains analyseurs sont allés jusqu'à utiliser les indices phonétiques pour décoder le texte imparfait de l'enfant.

Au niveau des traducteurs, l'expérience principale est celle qui a consisté à faire programmer les enfants en LOGO, dans le but d'explorer le corpus sous-tendu par les primitives notionnelles, ou dans le but de s'initier à la programmation.

Le Langage Androïde

Les outils dont l'enfant doit disposer pour créer son propre langage d'application en robotique éducative doivent être de trois ordres.

Premièrement, l'enfant doit disposer de primitives robotiques pour contrôler le robot et permettre à l'ordinateur d'obtenir de l'information sur le système robotique, par exemple les mots INTERRUPTEUR, CELLULE, COMPTEUR, pour les entrées et POSITIF, NÉGATIF et NUL, pour les sorties.

Deuxièmement, l'enfant doit disposer de conjonctions telles TANT QUE, SI, LORSQUE, qui sont autant d'outils dans l'expression de l'algorithme visant à résoudre un problème ou à définir un mot.

Troisièmement, l'enfant doit disposer de méta-mots dont la fonction est de permettre de définir d'autres mots. Les contraintes que nous avions, lorsque nous recherchions les meilleurs méta-mots, étaient multiples. D'une part, il était nécessaire qu'ils soient peu nombreux (moins de dix). D'autre part, ils se devaient d'être connus de l'élève et ils se devaient de suggérer correctement leur fonction. Enfin, nous désirions qu'ils soient puissants, c'est-à-dire qu'ils puissent engendrer une expression naturelle de la solution de nombreux problèmes en robotique éducative.

Les méta-mots dont nous avons convenu sont les suivants.

D'abord, les méta-mots standards: DÉFINITION, MOUVEMENT qui permettent de définir une procédure, et VARIABLE, qui permet de définir une structure de données.

Ensuite, les méta-mots sémantiques OBJET, SYNONYME, INSIGNIFIANT: OBJET permet de créer un réseau sémantique, SYNONYME crée un nouveau mot synonyme d'un mot connu, INSIGNIFIANT crée un mot qui n'a pas de fonction algorithmique, mais qui complète une expression.

Enfin, les méta-mots syntaxiques DÉTERMINANT, PRÉCÉDENT, SUIVANT: DÉTERMINANT permet de définir des expressions consacrées, dont nous savons qu'elles représentent une part importante du langage [1], PRÉCÉDENT et SUIVANT permettent de faire varier l'ordre des mots. En effet, l'expression AVANCE DE 3 CM peut être analysée de la façon suivante: l'argument de AVANCE est 3 CM, l'argument de CM est 3. Par conséquent, l'argument de AVANCE suit le mot, tandis que l'argument de CM précède le mot. Dans le langage dont la syntaxe est rigide, il faudrait écrire AVANCE CM 3, comme en LOGO, ou 3 CM AVANCE, comme en FORTH. Dans le langage Androïde, puisque le mot DE est «insignifiant», on peut définir et effectivement utiliser l'expression AVANCE DE 3 CM.

L'activité de l'enfant

Pour illustrer l'activité type d'un enfant qui utilise le Langage Androïde, reprenons le problème du début, et supposons que le problème se complique par l'adjonction d'une automobile rouge et d'une automobile verte.

Les phrases que l'enfant peut vouloir utiliser pour décrire le problème pourraient être les suivantes: «l'automobile rouge frappe à l'avant» ou «l'automobile verte frappe à l'avant». Le «frappe à l'avant» est utilisé de la même façon dans les deux phrases, malgré qu'il s'adresse à l'automobile rouge dans un cas et à l'automobile verte dans l'autre cas. L'enfant aura alors à définir un réseau sémantique simple associant les différents objets aux mots s'y rapportant (Figure 4).

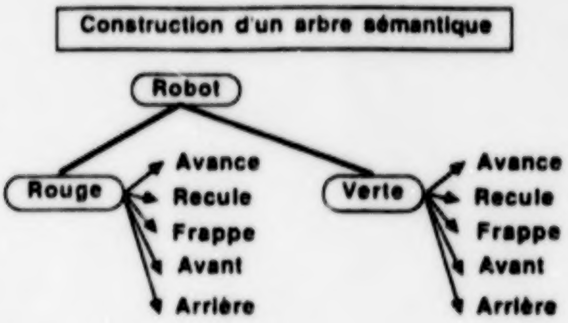


Figure 4

Ainsi, «frappe à l'avant» associé à l'automobile rouge pourra signifier «l'interrupteur 1 est pressé», tandis que «frappe à l'avant» associé à l'automobile verte pourra signifier «l'interrupteur 3 est pressé». Également, le mot AVANCE associé à l'objet ROUGE pourrait mettre en marche le moteur 1, tandis que le mot AVANCE associé à l'objet VERTE pourrait mettre en marche le moteur 2.

Par ailleurs, lors de la construction de l'algorithme, l'enfant se devra faire une étude syntaxique des phrases comme: «Si elle frappe à l'avant, elle recule.» (Figure 5). Cette analyse pourrait le conduire à comprendre que la condition est exprimée par la phrase «elle frappe à l'avant», que l'action est exprimée par la phrase «elle recule», que le délimiteur de la condition est la virgule et que le délimiteur de l'action est le point, que AVANT est l'argument de frappe, et que SI est la conjonction qui détermine la structure de phrase attendue.

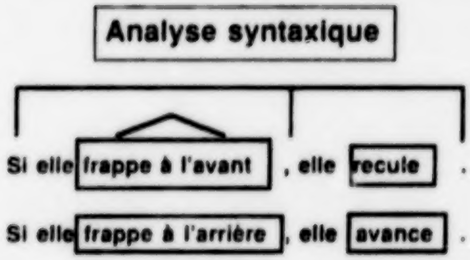


Figure 5

Un exemple

Un cycle complet de cette recherche a été tenté à l'école Louis-Joseph Papineau de la C.E.C.M. Après un an d'initiation à la robotique pédagogique, des élèves ont entrepris de

construire un théâtre robotisé de marionnettes contrôlé entièrement par ordinateur: le rideau, les décors, les réflecteurs et les marionnettes.

Après avoir construit les différents éléments de ce théâtre, il ont eu à créer un langage d'application permettant de contrôler le théâtre, de façon à ce que des élèves néophytes puissent éventuellement être appelés à contrôler ce théâtre. Ce langage d'application se devait d'être suffisamment naturel et simple pour que le temps d'apprentissage se limite à quelques minutes (en excluant l'initiation à l'ordinateur).

Le travail des élèves a consisté principalement en la construction de primitives «théâtrales» et de quelques expressions consacrées. En voici quelques exemples:

OUVRE LE RIDEAU, BAISSE LE DÉCOR 1, LÈVE LE DÉCOR 2.
ALLUME LE RÉFLECTEUR JAUNE, ÉTEINT LE RÉFLECTEUR ROUGE.
C'EST LE MATIN, C'EST LE SOIR, C'EST LA NUIT.
RAMON VA AU CENTRE, ZOÉ VA EN COULISSE, RAMON TOURNE À GAUCHE.
SIMULTANÉMENT ZOÉ TOURNE FACE AU PUBLIC, RAMON SE PENCHE.

Deux observations: premièrement, quelques petits scénarios ont été réalisés avec ce langage d'application, et pour le moment, il semble que ce soit suffisant; deuxièmement, après trois mois de vacances, pendant lesquels le théâtre a été inutilisé, les élèves se sont remis à programmer le théâtre comme s'ils n'avaient jamais cessé de le faire.

Conclusion

Nous avons remarqué qu'il s'établissait un cycle itératif entre la définition des mots et la définition des phrases et des expressions consacrées. Nous ne savons pas quelle en est la dynamique. À quelques reprises, des expressions qui venaient naturellement à l'esprit des enfants n'ont pu être implantées dans le langage. Nous ne savons pas quelle est la proportion des expressions possibles dans ce langage par rapport à l'ensemble des expressions naturelles. Certaines discussions entre les élèves et les chercheurs ont porté sur la structure des phrases et la façon dont une phrase particulière porte le sens. Nous ne savons pas quelle est la partie de cet exercice de construction d'un langage qui est «utilisable» dans l'enseignement de la langue naturelle.

Ce que nous savons, c'est que cette approche de la communication de l'enfant avec l'ordinateur représente un pas en avant dans la communication des **enfants entre eux**, grâce à l'introduction d'un langage compréhensible par tous les intervenants: l'ordinateur, les élèves et les professeurs.

BIBLIOGRAPHIE

[1] *Second Conference on Applied Natural Language Processing*, Février (1988). Austin, Texas, USA Association for Computational Linguistic.

[2] Winograd, T. (1983). *Langage as a Congnitive Process, Syntax*, Addison-Wesley Publishing Co.

[3] Shwartz, S.C. (1987). *Applied Natural Langage Processing*, Petrocelli Books, Princeton, N.J.

[4] Brachman, R.J., Lévesque, H.J. (1985). *Reading in Knowledge Representation*, Morgan Kufmann Publishers, Los Altos, Ca.

[5] Harris, M.D. (1985). *Natural Language Processing*, Reston Publishing Company Inc., Reston, Virginia.

[6] Minsky, M. (1981). *A framework for representing Knowledge*, MIT Press, Cambridge, MA

[7] Bobrow, D.G., Winograd, T. (1977). *An Overview of KRL, a Knowledge Representation Langage*, Cognitive Science, *1(1)*, pp. 3-46.

[8] La Palme, J.-B., Bélanger M. (1990). *Introduction à la Robotique éducative avec le Langage Androïde*, Les Éditions Micro-Éduc Inc., Montréal.

Des exemplaires du présent document sont en vente à l'adresse suivante:

Centre Interdisciplinaire de Recherche sur
l'Apprentissage et le Développement en Education
Université du Québec à Montréal
C.P. 8888, Succ. A,
Montréal, Québec
H3C 3P8

Publié par le Centre Interdisciplinaire de Recherche sur l'Apprentissage et le Développement en Education

Université du Québec à Montréal

Conception graphique: Pierrette [illegible]